
# ARTDAQ WORKSHOP

Karol Hennessy, Giovanna Lehmann Miotto

January 12, 2018

University of Liverpool

# THANKS FOR THE

- welcome
- preparation (great talks and content)
- valuable discussion
- overall commitment
- hospitality ☺

# BOARDREADER

- Functionality is mostly there
- Interface to get request information will be provided
  - API
- Snapshotting
  - How do we respond to status commands?
- API to publish GOOD/BAD Inhibit status
- Tracing and logging
  - We need to educate the BR devs on the use of TRACE
  - Agree on naming conventions
  - Support Wildcards for TRACE Names.
- reception of data from HW should not be coupled with getNext() (independent threads)
- Metric monitoring essential but not at full potential right now
- BoardReader shouldn't generate empty events for "missing" seq-IDs
  - i.e., BR should not have any assumption of sequentiality of requests

# INHIBIT MASTER

- Publish/Subscribe mechanism seems correct
- Eventually need data on DIM, so perhaps do everything with DIM
- To be implemented by all BRs
- Wes to contact Phil on logics in Timing BR. IM perhaps sits in Timing codebase
- NOTE: without DFO - hard to identify origin of backpressure
- mechanism to view DIM metrics remotely (i.e., FNAL)

- Switch from MPI to TCP and test with current v2.03.04
- We still consider a complete DFO a requirement for ProtoDUNE-SP
  - timing BR provides seq-ID and timestamp to DFO (timing BR behaves like any other BR w.r.t fragment sending)
  - Assignment of destination when a trigger arrives
  - Distribution of the timestamp,dest,seq-ID to
    - in push mode : BRs
    - in pull mode : only the affected EB node (when requesting events, EB will provide all info to BRs)
  - Bookkeeping of tokens dealt with by DFO
    - completion of event communicated by EB
  - DFO is an Inhibit source indicating backpressure from EB or DataLogger
- If the workload for this component is too much for the artDAQ team, we will endeavour to make it ProtoDUNE-SP specific and find the required effort

- Push-Pull
  - Pull is only useful for traffic shaping purposes
  - If as artDAQ experts, you are confident that no traffic shaping is needed, we are happy staying with the push model
    - This implies the destination assignment contains seq-ID and timestamp
    - Our preference is for the dest. assignment be done via TCP (rather than broadcast). This also gives us the option to vary dest. assignments due to unforeseen congestion

# DATARECEIVER (EB, DL, DISP, ETC)

- v3 DataReceiver - can merge DataLogger & EventBuilder dependent on desired topology (needs testing)
- remove strict ordering of sequence IDs ? “There are no missing IDs only requested IDs”
  - does art still require sequential IDs?
- TCP receiver point to point is not a constraint (no request for Dispatcher to multiple OMs)
- Start performance tests on data file writing

- Functionality is complete for PDSP
- Minor feature requests
  - add comment to config tag
  - and a timestamp
  - invalidated/deleted configs
- Bug fix - no clobbering of existing configs
- Trigger config will be another configuration
  - Trigger BR will access it and use it during START transition
  - How to include in data files to be explored
  - Every trigger config change is a new run
    - STOP/START must be fast ($< 20$s)
  - Shifter will select trigger config

- DAQ Interface is functional now
- Aim to restrict it to Conf Mgmt tasks for ProtoDUNE-SP
  - Retain full DAQ Interface functionality for testing
- JCOP to take over process control
  - ensure we provide correct env to procs
  - procs not running with MPI $\Rightarrow$ proc dependencies must be correctly handled
- Partitioning
  - one instance of DAQ Interface per partition
  - parameterise port num

- Logging - check timescale for art rewrite of Message Facility (compliant with logstash)
- Config dump - with default params
- Releases
  - nightly merge tests with feature branches ?
  - happy with continued support by John as librarian
- Agreed to have an artDAQ on-call expert at beam time

# TIMESCALES

- Imminent
  - testing of TCP
  - some fixes on metric mon.
  - logging in logstash (if possible?)
  - bug fix of config clobbering
  - availability of “Mains” to test without MPI
- v3.0 - end Jan
  - DataReceivers
    - Refactorisation of DL, EBs, Disp.
  - applications identified by name (logging, metric mon, inhibit master)
  - DAQInterface without proc. mgmt.
  - JCOP proc. mgmt (end Feb)
  - complete functionality of metric monitoring and log formatting
  - Inhibit Master in ProtoDUNE-SP context (end Feb)
  - additional feature in conf. mgmt.

# TIMESCALES

- v3.1 - end Mar?
  - handling of trigger configs
  - Unicast control messages (dest. assignments, frag. requests)
  - DFO
  - Exercise partitioning
- v3.2 - end May?
  - Fault tolerance testing